In the Claims:

1. (Original) A method of communicating over a controller area network (CAN) bus, comprising:

routing registration information from a plurality of processor-enabled peripheral devices to a controlling software component;

routing a periodic heartbeat message from the controlling software component to the plurality of processor-enabled peripheral devices to enable each of the plurality of processor-enabled peripheral devices to maintain its registered status; and

if necessary, routing messages from the controlling software component to one or more of the plurality of processor-enabled peripheral devices on a discrete basis over the CAN bus to control the one or more of the plurality of processor-enabled peripheral devices.

2. (Original) The method of claim 1, further comprising causing the controlling software component to consecutively receive frames of a multi-frame message transmitted from one of the plurality of processor-enabled peripheral devices.

3. (Original) The method of claim 1, wherein the routing of messages from the controlling software component to one or more of the plurality of processor-enabled peripheral devices on a discrete basis to control the one or more of the plurality of processor-enabled peripheral devices comprises routing messages each having a like header to the one or more of the plurality of processor-enabled peripheral devices on a discrete basis to control the one or more of the plurality of processor-enabled peripheral devices.

4. (Original) The method of claim 3, wherein the routing of messages each having a like header to one or more of the plurality of processor-enabled peripheral devices on a discrete basis to control the one or more of the plurality of processor-enabled peripheral devices comprises routing messages each having a common header component and a CAN header component to the one or more of the plurality of processor-enabled peripheral devices on a discrete basis to control the one or more of the plurality of processor-enabled peripheral devices.

5. (Original) The method of claim 4, wherein the routing of messages each having a common header component and a CAN header component to the one or more of the plurality of processor-enabled peripheral devices on a discrete basis to control the one or more of the plurality of processor-enabled peripheral devices further comprises routing messages each having a common header component and a CAN header component without specific knowledge by the controlling software component of the CAN header component.

6. (Original) A method of communicating over a controller area network (CAN) bus, comprising:

routing a registration message from a processor-enabled peripheral device to a controlling software component;

at the processor-enabled peripheral device, periodically receiving a heartbeat message from the controlling software component subsequent to the routing of a registration message from a processor-enabled peripheral device to a controlling software component; and

receiving at the processor-enabled peripheral device discrete control messages that are transmitted from the controlling software component.

7. (Original) The method of claim 6, wherein the receiving at the processor-enabled peripheral device discrete control messages that are transmitted from the controlling software component comprises filtering the transmitted control messages at the processor-enabled peripheral device to enable only the discrete control messages intended specifically for the processor-enabled peripheral device to reach the processor-enabled peripheral device.

8. (Original) The method of claim 7, wherein the filtering of the transmitted control messages at the processor-enabled peripheral device to enable only the discrete control messages intended specifically for the processor-enabled peripheral device to reach the processor-enabled peripheral device comprises filtering the transmitted control messages at the processor-enabled peripheral device via a hardware filter to determine whether the transmitted control messages are for a certain type of processor-controlled peripheral device, and filtering the transmitted control messages at the processor-enabled peripheral device via a software filter to determine processor-controlled peripheral device numbers from respective message CAN headers.

9. (Original) The method of claim 8, further comprising receiving at the processor-enabled peripheral device all message frames following the processor-enabled peripheral device type and number information subsequent to the filtering of processor-enabled peripheral device type and number information from the discrete control messages intended specifically for the processor-enabled peripheral device.

10. (Original) The method of claim 7, wherein the filtering the transmitted control messages at the processor-enabled peripheral device to enable only the discrete control messages intended specifically for the processor-enabled peripheral device to reach the processor-enabled peripheral device is invisible with respect to the controlling software component.

11. (Original) The method of claim 6, further comprising, at the processor-enabled peripheral device, consecutively receiving frames of a multi-frame discrete control message.

12. (Original) A controller area network (CAN) bus for enabling a controlling software component to communicate discretely with each of a plurality of processor-enabled peripheral devices irrespective of whether the processor-enabled peripheral devices are like devices, comprising:

a processor for routing control messages between the controlling software component and the plurality of processor-enabled peripheral devices;

a plurality of bus lines for connecting the processor to the controlling software component and the plurality of processor-enabled peripheral devices; and

the processor for enabling the control messages to be discretely transmitted from the controlling software component to one or more of the plurality of processor-enabled peripheral devices.

13. (Original) The CAN bus of claim 12, wherein the processor is programmed with a software switch for enabling the controlling software component to consecutively receive frames

of a multi-frame message transmitted from one of the plurality of processor-enabled peripheral devices.

14. (Original) The CAN bus of claim 12, wherein the processor is programmed for enabling transmission of multi-frame CAN bus messages.

15. (Original) The CAN bus of claim 12, wherein the processor is further for generating a CAN header component for each of the control messages transmitted from the controlling software component to enable the control messages to be discretely transmitted from the controlling software component to one or more of the plurality of processor-enabled peripheral devices.

16. (Original) The CAN bus of claim 12, wherein the processor is further for causing frames of a multi-frame message transmitted to one of the plurality of processor-enabled peripheral devices from the controlling software component to be consecutively received at the one of the plurality of processor-enabled peripheral devices.

17. (Original) The CAN bus of claim 12, wherein the processor and the plurality of bus lines are implemented on a controlling board of a wireless base station.

18-27 (cancelled)